



Bombay Stock Exchange's Release 1.1

BSE's Enhanced Order Book Interface Manual

Version	V1.1
Date	9. Februar 2015

Bombay Stock Exchange's Release 1.1

BSE India Pvt. Ltd

BSE's Enhanced Order Book Interface

V1.1

Strictly for private circulation only. This document must not be circulated to other users without prior permission of BSE.

Contents

1	List of Abbreviations	4
2	Introduction	5
2.1	Purpose of this document	6
2.2	Document Outline	6
2.3	Further Reading Material	6
3	Characteristics	7
3.1	Functional Characteristics	7
3.2	Technical Characteristics	7
4	Order Book Management	9
4.1	Building the Order Book	10
4.2	Adding an Order	11
4.3	Identifying an Order	12
4.4	Modifying an Order	12
4.5	Deleting an Order	12
4.6	Order Executions	12
4.7	Trade Statistics	13
4.8	Auctions	13
4.9	Product and Instrument States	13
4.10	Heartbeats	14
4.11	Recovery	14
5	Availability of Enhanced Order Book Service	16
6	Message Formats	18
6.1	Datagram Structure	18
6.2	Incremental Messages	19
6.3	Snapshot Messages	20
6.4	Message Overview	23
6.5	Data Types	23
7	Message Layout	24
7.1	Overview of Supported Message Types	24
	General	24
	Snapshot	24
	Order Data	24
	Trade Data	24
	State Change	25
7.2	General	26
	Packet Header	26
	Heartbeat	28
7.3	Snapshot	29
	Product Summary	29
	Snapshot Order	31
	Instrument Summary	32
7.4	Order Data	35

Auction Clearing Price	35
Order Add	36
Order Modify	37
Order Modify Same Priority	39
Order Delete	41
Order Mass Delete	42
Partial Order Execution	43
Full Order Execution	44
7.5 Trade Data	45
Execution Summary	45
Instrument Info	47
7.6 State Change	48
Product State Change	48
Instrument State Change	50
8 Appendix	52
8.1 Product Scope	52
8.2 Synthetic Prices	52
8.3 Connecting BSE EOBI and BSE ETI data	53
8.4 Multicast addresses	54
8.5 Reference data for BSE EOBI	54
9 Change log	55

1 List of Abbreviations

The following are the abbreviations and definitions used in this manual:

BSE EOB	BSE E nhanced O rders B ook I nterface
BSE EMD	BSE E nhanced M arket D ata I nterface
BSE ET	BSE E nhanced T rading I nterface
BSE MD	BSE M arket D ata I nterface
BSE RD	BSE R eference D ata I nterface
BSE RDF	BSE R eference D ata F ile
FIX	F inancial I nformation eX change. The Financial Information eXchange ("FIX") Protocol is a series of messaging specifications for the electronic communication of trade-related messages.
Out-of-Band	Incremental-messages and Snapshot-messages are delivered on different multicast channels.
Live - Live	Concept whereby data is disseminated simultaneously via two separate channels called "Service A" and "Service B".
BBO	B est B id and O ffer (can refer to price and size).
Match Step	Product-wide day-unique identifier for each price level of the match event.
Potential Auction Price	If the order book becomes crossed during an auction, then a potential auction price is formed and communicated to all participants.

This manual uses conventions to highlight certain words and phrases and draw attention to specific pieces of information.

Therefore, all message names related to BSE Enhanced Order Book Interface feeds are in fixed width font like this and all field names are in *italic* to separate them from ordinary text. **Bold** highlighting will be used when a new term is introduced, or to emphasize the importance of a word.

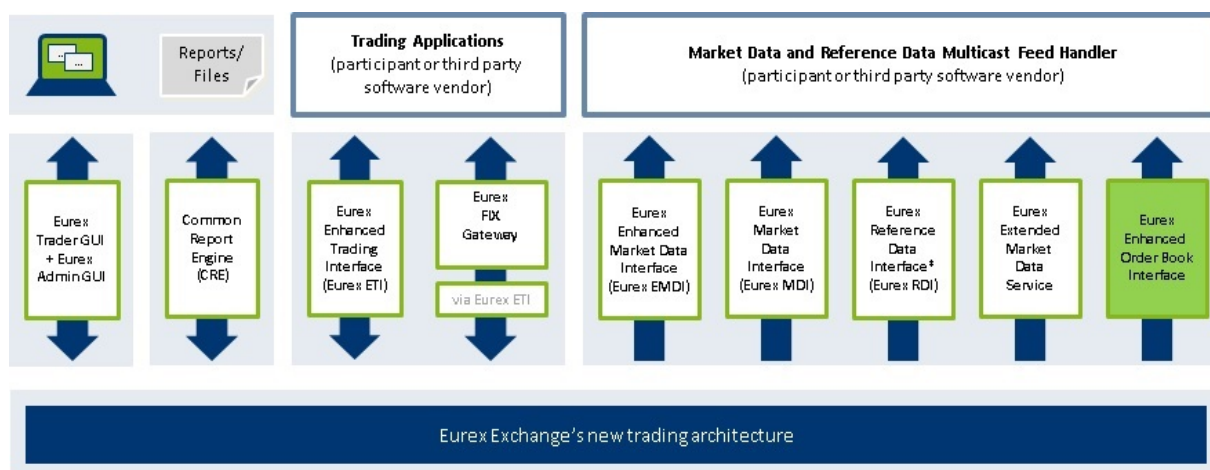
2 Introduction

The new **BSE Enhanced Order Book Interface (BSE EOBI)** provides the entire visible order book, by publishing information on each individual order, along with executions and state information in real-time and in an un-netted manner. The new interface provides an additional alternative to recipients of the BSE Enhanced Market Data Interface (BSE EMDI).

Though most of the functional concepts used are similar to those of BSE EMDI, the new interface provides greater transparency and efficiency, together with a high throughput at minimal latency. The BSE EOBI disseminates public market data with the following features:

- A full order depth feed; there is no depth restriction.
- Information is sent in form of fixed-length binary messages.
- Intelligent packing of messages into a datagram by including repetitive entities only once in a message.
- Utilization of the widely adopted FIX standard to decrease integration efforts and on-going support costs.
- Corresponding reference data information is available via the existing BSE Reference Data Interface (BSE RDI) and the BSE Reference Data Files (BSE RDF).
- Dissemination of incremental messages (following state changes) and all Snapshot messages follow a publishing in sequence based on:
 1. Side (bid first, offer second),
 2. Price (best price first),
 3. Time (highest time-priority first).

The BSE EOBI extends the current interface landscape of the BSE Exchange's T7 with regard to market and reference data interfaces.



Picture 1: Interface Landscape of BSE's T7

As depicted above, the new interface provides an additional market data interface alongside the existing BSE EMDI and BSE MDI interfaces.

The BSE EOBI is designed for participants that rely on **low-latency** at a high throughput with a **high band-width network**. The interface disseminates all visible orders without any depth restriction, when the order books are open, along with order executions and state information via incremental messages in un-netted manner. Furthermore, snapshot messages always carry existing visible orders without any depth restriction at the time of sending.

Multicast address and port combinations of BSE EOBI will be different from BSE EMDI and BSE MDI.

2.1 Purpose of this document

The purpose of this document is to provide guidance for programmers developing applications that receive public market data from the BSE EOBI feeds.

It covers a complete reference, describes the general business behavior and provides concepts for the implementation.

The most recent version is available at:

www.bseindia.com/nta.aspx > Market Data

2.2 Document Outline

The following chapter, Chapter 3 - "Characteristics" gives an overview of the functional and technical features of the new BSE Enhanced Order Book Interface.

Chapter 4 - "Order Book Management" outlines the availability of messages, the initial build-up of the order book, and the processing of order book updates.

Chapter 5 - "Availability of Enhanced Order Book Service" presents the availability of the Enhanced Order Book Interface according to the state of trading during the day.

Chapter 6 - "Message Formats" outlines the general structure of messages sent out over the BSE Enhanced Order Book Interface, followed by the specific individual message layouts in Chapter 7.6 - "Message Layout".

Chapter 8 - "Appendix" describes:

- the product scope of this interface, (see Appendix 8.1 - "Product Scope"),
- where synthetic pricing information can be found, (see Appendix 8.2 - "Synthetic prices"),
- how public market data and private data can be synchronized between BSE EOBI and BSE Enhanced Transaction Interface (BSE ETI). See Appendix 8.3 - "Connecting BSE EOBI and BSE ETI data",
- how the reference data can be extracted from BSE RDI and/or BSE RDF (see Appendix 8.5 - "Reference data for BSE EOBI").

2.3 Further Reading Material

BSE recommends participants to be familiar with the concepts described in the following documents:

- BSE Market and Reference Data Interfaces Manual
- BSE Trading Interface Manual

BSE related documents are available at:

www.bseindia.com/nta.aspx

FIX-messages and FIX-tag related information is available at:

www.fixprotocol.org > Specifications

www.fixprotocol.org > FIXimate3.0.

3 Characteristics

The BSE EOBI is based on the same concepts as the BSE EMDI. However, there are some functional and technical characteristics that distinguish the BSE EOBI from BSE EMDI.

3.1 Functional Characteristics

The BSE EOBI disseminates:

- The instrument identifier, side, price, priority timestamp and quantity of each visible order side.
- Trade price and traded quantity for each executed on-exchange trade.
- Order book information disseminated without any depth limitation.
- The trading status of each product and corresponding instruments.
- Recovery via BSE EOBI snapshots.

Each order can be **uniquely** identified by the combination of instrument identifier, side and priority timestamp.

In order to send public market data as fast as possible, the BSE EOBI publishes only very specific market information. However, participants can derive certain information themselves based on the messages sent out by the BSE EOBI. The following information is not explicitly provided, however can be derived, if needed.

- Price levels; can be derived from individual orders.
- Aggregation at price levels; can be derived from individual orders.
- Information about synthetic prices; can be derived from visible orders received on the BSE EOBI feed.
- Fully matched incoming visible orders; can be derived from execution messages.
- Trade statistics are not provided via the incremental channel to keep the size of messages as small as possible. They can be derived from the order execution messages sent out on the BSE EOBI incremental channel. But, on the other hand, trade statistics are sent out on the BSE EOBI snapshot channel for recovery purposes.

3.2 Technical Characteristics

The BSE EOBI contains similar technical characteristics as the BSE EMDI, such as “Live - Live” multicast, distribution mode and sequence numbering schemes. Anticipating a high load, the size of messages is kept as small as possible.

The following are highlights of the technical characteristics of the BSE EOBI :

- Low-latency multicast for data dissemination with “Live - Live” concept.
- Fixed length optimized message layouts without any compression.
- Uses push-based publishing model in Out-Of-Band distribution mode.
- Packet and message sequence number schemes (same as on the BSE EMDI feed). However, as opposed to BSE EMDI/MDI/RDI, the field *MarketSegmentID*, i.e., the product identifier will be used instead of *SenderCompID* in packet header. Therefore, the *SenderCompID* will be absent in both the Packet Header and the Message Header.
- Matching Engine-In timestamp, *TrdRegTSTimeIn*, as part of orders.
- Little Endian and basic data types are used.
- Message padding for better byte alignment.
- Recovery via BSE EOBI snapshot channel as similar to BSE EMDI.

All messages are designed to be as small as possible and are following FIX 5.0 SP2 semantics. The maximum number of bytes per transmission unit (MTU) is limited to 1372 bytes.

The rule for the **distribution sequence** across partitions is as follows:

Even partitions : Publish on Service A first, then on Service B.

Odd partitions : Publish on Service B first, then on Service A.

The above rule is applied by using the field *PartitionID*. It is available in the Product Snapshot message via the BSE RDI and BSE RDF, master files and in the packet header of BSE EOBI.

All functional and technical reference data information needed for the BSE EOBI is provided by the BSE RDI and/or BSE RDF/Master files, similar to the current procedure for the BSE EMDI. The multicast addresses and ports for both services are disseminated in the product reference information. Multicast addresses and port information don't change during trading hours. See [8.5](#) - "Reference data for BSE EOBI" .

4 Order Book Management

The BSE EOBI provides an **explicit** message for each order book update by publishing the instrument identifier, side, price, displayed quantity, priority timestamp and “matching engine in” timestamp of each visible order in the entire order book, along with the order execution and state information. As described earlier, each order is uniquely identified by the combination instrument identifier, side and priority timestamp.

An outline of the **visibility** of orders on the BSE EOBI is shown below:

Order Type	Visible in Order book
Regular Limit Order	yes
Triggered Order – Stop Limit Order	yes
Market Orders	no
Stop Market Order (un-triggered)	no
Stop Limit Order (un-triggered)	no
Regular Limit Order – IOC	no
All types of Rejected Orders	no

Table 1 - Visibility of orders on the BSE EOBI

For each instrument within a product, snapshot messages can be received via the BSE EOBI snapshot channel to build the initial order book. Once the initial order book is built, the order book must be maintained using the corresponding order book updates received on the BSE EOBI incremental channel. On the BSE EOBI incremental channel, order messages are used by participants to maintain the order book, while explicit state change messages are provided to communicate current product and instrument state. Intra-day complex instrument changes will also be communicated via the BSE EOBI incremental channel.

To assist fine filtering and error discovery on the participant side, the BSE EOBI keeps messages in line using a multi-sequencing paradigm. It uses the following two sequencing methods: **packet sequence number** and **message sequence number**.

Packet Sequencing

Each packet on the BSE EOBI feeds is sequenced using contiguous packet sequence numbers. The packet sequence number is incremented for each packet across products on the same feed.

Message Sequencing

In addition to packet sequencing, each product on the BSE EOBI feeds is sequenced contiguously by using message sequencing. This should allow participants to filter products of interest only. The message sequence number is incremented per product across the different message types.

The following sections describe the order book management with respect to the messages sent over the BSE EOBI.

Message layouts can be identified by the *templateID* field which is the (exchange wide) unique identifier for the message layout, and is included in each Message Header. The *templateID* also determines the fixed size of the message.

Message	Template ID
Order Add	13100
Order Modify	13101

Order Modify Same Priority	13106
Order Delete	13102
Order Mass Delete	13103
Partial Order Execution	13105
Full Order Execution	13104
Execution Summary	13202
Auction Clearing Price	13501
Product State Change	13300
Instrument State Change	13301
Product Summary	13600
Instrument Summary	13601
Snapshot Order	13602
Heartbeat	13001
Instrument Info	13203

Table 2 - BSE EOBI messages with assigned template IDs

4.1 Building the Order Book

Product and instrument reference data information required to process the BSE EOBI market data is provided by the BSE RDI and/or BSE RDF, similar to the current procedure for BSE EMDI, also see Appendix See [8.5](#) - "Reference data for BSE EOBI".

Messages in the BSE EOBI snapshot channels are grouped by product. In order to build an initial order book, participants subscribe to the BSE EOBI snapshot channel. The content of one **snapshot cycle** for one product is denoted in Figure 5 (see Snapshot Messages). The individual orders in the order book are represented in the snapshot message using the Snapshot Order messages. The snapshot messages contain the field *LastMsgSeqNumProcessed* to enable participant synchronization between the BSE EOBI snapshot channel and the BSE EOBI incremental channel.

While subscribed to the BSE EOBI snapshot channel, participants should keep processing incoming data from the BSE EOBI incremental channel. Any incoming incremental messages with a sequence number higher than the *LastMsgSeqNumProcessed* received in the snapshot message should be applied to the order book after the full snapshot message is processed.

The following data is provided via the BSE EOBI snapshot channel:

- Product State information,
- Instrument State information,
- Trade Statistics per instrument,
- All visible orders in the order book.

During the Continuous Trading instrument state, all visible orders in the order book will be published on the BSE EOBI incremental channel.

During the Auction instrument state, the BSE EOBI incremental channel will broadcast the auction clearing price (indicative auction price) if book is crossed. After the auction phase, trades that took place during the auction will be published using the Execution Summary messages, before the corresponding state changes.

The sequencing of the data in a snapshot cycle is based on the product identifier, the instrument identifier and on the price level. For the product and instrument identifier, the **sending order sequence** is ascending and the orders are sorted from best to worst prices (buy orders are sorted from highest to lowest, and sell orders from lowest to highest).

The visible orders are sent alternating between buy and sell sides, where orders at the same price level are sorted by order time priority from the oldest to the newest order. The visible order book is disseminated per price level in a zig-zag manner, meaning both the sides (Bid and Offer) at each price level are disseminated before moving on to the next price level. If one side providing more orders on the same price level as the opposite side, all orders of the same price level are processed before switching to the next price level.

Assuming the following arbitrary order book is sorted according to imaginary order priority timestamps and order prices where in the orders with the same order prices are sorted according to imaginary order priority timestamps.

Buy	Sell
Order _{B1} 100.05	Order _{S1} 100.50
Order _{B2} 100.05	Order _{S2} 100.55
Order _{B3} 99.95	Order _{S3} 100.55
Order _{B4} 99.90	Order _{S4} 100.55
Order _{B5} 99.00	Order _{S5} 101.00
Order _{B6} 97.00	

Picture 2: Order book in a zig-zag manner

As it can be seen from table above, the orders denoted by B1, B2 and S1 are on the first price level. The orders denoted by B3, S2, S3 and S4 are on the second price level. The orders B4 and S5 are on the third price level. In price level fourth and fifth buy orders exists only.

The resulting sending order sequence in zig-zag fashion is: B1, S1 and B2, B3, S2, S3, S4, B4, S5, B5 and B6.

4.2 Adding an Order

An Order Add message will be sent each time a visible order is added to the order book of the corresponding instrument. The message includes the instrument identifier, priority timestamp, side, price, displayed quantity of the order and its matching engine in timestamp.

The Order Add message includes among other the priority timestamp and side, which are to be used as the instrument-wide **unique identifier** of this order, as long as the order is not modified. See 4.4 - "Modifying an Order". The instrument identifier, priority timestamp and side will be the reference key for all future updates for the order. See 4.3 - "Identifying an Order".

Information about an incoming order, that matched fully against to one or more orders in the order book, can be derived from the associated execution messages or execution summary only.

The remaining part of an incoming order that matches partially will be reported with an Order Add message after all associated executions.

The Order Add messages also include the "matching engine in" timestamp (conveyed by *TrdRegTSTimeIn*) of the order, which conveys when the corresponding order transaction has been received by the matching engine. Please note that *TrdRegTSTimeIn* will not be set in case of a self triggered transaction (without any external actor). For ex. a state change resulting in opening of the book.

4.3 Identifying an Order

Participants are able to identify their own orders on the BSE EOBI by using the unique identifier, the priority timestamp and side, as stated earlier.

In order to provide participants with the priority timestamp of the orders, the field *TrdRegTSTimePriority*, will be provided in the Order messages of the BSE EOBI and in the BSE ETI responses.

In order to identify matching of own orders, the priority timestamp of the order and a unique match step identifier for each price level of the match event will be provided, i.e., the fields *TrdRegTSTimePriority* and *TrdMatchID* in the execution messages in the BSE EOBI will correspond to the field *TrdRegTSTimePriority* and *FillMatchID* and/or *QuoteEventMatchID* in the BSE ETI (See Appendix 8.3 - "Connecting BSE EOBI and BSE ETI data").

4.4 Modifying an Order

If the time-priority, price and/or displayed quantity of an existing order changes, then an Order Modify message will be sent.

A modification might result in the order being assigned a new priority timestamp (for example, in the case of a price modification). If its the case, then the *TrdRegTSPrevTimePriority* field will be given in the Order Modify message and the new priority timestamp will be available in the Order Modify message. The new *TrdRegTSTimePriority* along with the side will be the new unique key for the order in the future.

However if there is no priority loss with the modification (which may occur for example when quantity is reduced) then the Order Modify Same Priority message will be sent and *TrdRegTSTimePriority* field will contain the original order priority-timestamp.

"Matching engine in" timestamp, *TrdRegTSTimeIn*, of the order will be amended accordingly.

4.5 Deleting an Order

When an order is deleted, the BSE EOBI will publish the instrument identifier, the priority timestamp of the order, side, price and transaction time, i.e., the fields *SecurityID*, *TrdRegTSTimePriority*, *Side*, *Price* and *TransactTime*, which will enable participants to quickly identify and delete the corresponding order from the order book. The "matching engine in" timestamp of delete request, *TrdRegTSTimeIn*, will be provided as well. Please note that *TrdRegTSTimeIn* will not be set in case of a self triggered transaction (without any external actor). For ex. an instrument expiry causing an order book clean up.

4.6 Order Executions

In order to ease the processing of matches along with the other order book updates by participants the following information is disseminated for each match:

- first, an execution summary message will be sent when an incoming order has been matched against orders that were already in the order book,
- second, messages that convey the individual executions of visible orders are published¹.

¹That implies individual executions are not sent of an incoming order matches against non-visible orders, i.e., if an incoming order matches against a market order or if a synthetic match happens.

The Execution Summary message contains the instrument identifier, side, aggressor time-stamp, worst price, total executed quantity, resting hidden quantity (if any) and match-time information of the incoming order.

For conveying the individual executions of the visible orders two template messages will be used for fully and partially executed orders.

In a sense, the Execution Summary messages can be used by participants for fast trading decisions. The individual order execution messages should be used by participants for order book maintenance to ensure the correctness of the order book.

The order execution messages will be sent whenever a visible order is **fully** or **partially** executed at its displayed price. Each **match step** will include a **product-wide day-unique identifier** of the trade, represented by the field *TrdMatchID*. This field will always have a value in the execution messages for a full or partial execution. The same unique identifier of the trade is made available to participants by the BSE ETI.

If the incoming order has been partially executed, then the remaining quantity will be reported with an Order Add message after all associated individual executions have been provided.

Triggered Stop Market orders or Stop Limit orders are reported like incoming Market or Limit orders, respectively.

4.7 Trade Statistics

Instrument trade statistics such as opening, closing, daily low and high prices are available via the BSE EOBI snapshot messages only. They are provided to participants for recovery purposes and are published included in the Instrument Summary message on the BSE EOBI snapshot channel. By design, they are provided as a repeating group as part of the Instrument Summary message and are not cut off.

When subscribed to the BSE EOBI incremental channel, participants can derive order book and trade statistics by combining the information received via the order and execution messages.

4.8 Auctions

When an instrument goes to an auction state, an Instrument State Change message is immediately published via the incremental channel.

Potential Auction price and quantity is published through Auction Clearing Price messages, which will carry the potential auction price for crossed order books respectively.

On the snapshot channels, the auction instrument state is reflected in the Instrument Summary message along with the trade statistics.

When an auction closes, i.e., an instrument leaving an auction, an Instrument State Change message is immediately published via BSE EOBI incremental channel followed by an optional Execution Summary message from the auction phase.

4.9 Product and Instrument States

In a Product State Change message, the product state can normally be found in the field *TradingSessionSubID*. Only for quiescent product states, the field *TradingSessionID* must be evaluated additionally to determine the actual product state.

A Halt state is additionally indicated by the field *TradSesStatus* containing the value "1 = Halted".

The instrument state is published with an *InstrumentStateChange* message and can be found directly in the field *SecurityTradingStatus*.

The status of the instrument (as opposed to the instrument state) distinguishes active and suspended and inactive instruments and is contained in the field *SecurityStatus*.

4.10 Heartbeats

Functional heartbeat messages, *Heartbeat*, are sent at a regular interval for less active products on the BSE EOBI incremental channels. A functional heartbeat message provides the message sequence number last sent in the field *LastMsgSeqNumProcessed* to allow participants to identify potential gaps. Heartbeats will be sent out as of the product state "Start-Of-Day".

Technical heartbeats will be provided on the specific ports assigned to technical heartbeat messages.

4.11 Recovery

Due to the unreliable nature of UDP multicast, UDP packets may be duplicated, delayed, missing, or arrive in an incorrect sequence. Therefore, the BSE EOBI uses "Live - Live" concept, as in BSE EMDI, for recovery purposes. Both the live feeds (A and B) are sequenced identically and participants should ideally process both feeds to detect data losses at an early stage.

If a packet is lost on both (Live - Live) feeds of the BSE EOBI incremental channel, then participants can take advantage of the **out-of-band** nature of the BSE EOBI. Participants can utilize the BSE EOBI snapshot channel to obtain the corresponding lost information, i.e., rebuild the initial order book, determine trade statistics and instrument states. For recovery, participants should recover on a product level (i.e., for all instruments of one product), for two reasons:

- The field *LastMsgSeqNumProcessed* in the snapshot cycle is given on product level, so in order to synchronize the BSE EOBI snapshot channel and the BSE EOBI incremental channel, participants should recover for all instruments in the product.
- Given the fact that there is no explicit information on synthetic price and quantity in the BSE EOBI, participants will have to re-determine the order books of all instruments to derive this information.

Participant Fail-Over

In the event of a market data fail-over, recovery on the participant side can be achieved by recovering the order book information via the BSE EOBI snapshot channel.

The BSE EOBI snapshot channel is synchronized with the BSE EOBI incremental channel through the use of message sequence numbering. Participants should subscribe to the BSE Order Book Snapshot channel while buffering incoming messages from the BSE EOBI incremental channel. Any incoming message from the BSE EOBI incremental channel with a *MsgSeqNum* higher than the value of the *LastMsgSeqNumProcessed* field received in the snapshot message should be applied to the order books after the full Snapshot message is processed.

Exchange Fail-Over

A fail-over of the BSE EOBI incremental channel will be recognizable by the following features:

- The *App/SeqNum* in the Packet Header is reset to 1.
- The *MsgSeqNum* in the Message Header will continue to be incremented contiguously (Ideally without any gap).

When a participant receives packets on a specific multicast address with an unexpected (lesser or equal) sequence number (either on service A or service B), it is advised, that the participant subscribes to the BSE EOBI snapshot channel again to rebuild the order book. Note that, because of the unreliable nature of the UDP protocol, packets may arrive out of sequence.

For a full restart of the BSE EOBI service, the BSE EOBI incremental channel will have the same features as the BSE EOBI snapshot channel:

- The *App/SeqNum* in the Packet Header is reset to 1.
- The *MsgSeqNum* in the Message Header is reset to 1.

In case of a full restart of the BSE EOBI service, participants must wait for the first message after the restart to be certain that a restart was executed. It is expected, that a full restart of BSE EOBI feed will take much longer than the configured heartbeat interval.

The field *App/SeqResetIndicator* is always set in the Packet Header of the first incremental messages after a (re-)start.

5 Availability of Enhanced Order Book Service

The BSE EOBI is available during the entire business day between product states "Start-Of-Day" and "End-Of-Day".

Table 4 below shows the information typically sent on the BSE EOBI during each product state. The messages listed in the table should serve as a super-set of messages and inform participants on "what-to expect" during each product state. However, it does not state any deterministic behaviour and should only be used as a guideline. The actual message set could be a sub-set of the listed messages depending on market conditions.

Product State	Messages
Start-Of-Day	Product State Change, Instrument State Change, Product Summary, Instrument Summary (incl. Trade Statistics), Heartbeat
Pre-Trading	Product State Change, Instrument State Change, Order Mass Delete, Product Summary, Instrument Summary (incl. Trade Statistics), Heartbeat
Trading	Product State Change, Instrument State Change, Add Order, Modify Order, Modify Order Same Priority, Delete Order, Partial Order Execution , Full Order Execution, Execution Summary, Auction Clearing Price (during Auction), Heartbeat, Product Summary, Instrument Summary (incl. Trade Statistics), Instrument Info, Snapshot Order,
Closing	Product State Change, Instrument State Change, Product Summary, Instrument Summary, (incl. Trade Statistics), Heartbeat, Trade Reversal
Post-Trading	Product State Change, Instrument State Change, Order Mass Delete, Product Summary, Instrument Summary (incl. Trade Statistics), Top Of Book, Heartbeat
End-Of-Day	Product State Change, Instrument State Change, Product Summary, Instrument Summary (incl. Trade Statistics), Top Of Book, Heartbeat
Post-End-Of-Day	-
Halt	Product State Change, Instrument State Change, Order Mass Delete, Product Summary, Instrument Summary (incl. Trade Statistics)

Bombay Stock Exchange's Release 1.1

BSE India Pvt. Ltd

BSE's Enhanced Order Book Interface

V1.1

Holiday	Product State Change, Instrument State Change, Product Summary, Instrument Summary (incl. Trade Statistics), Heartbeat
---------	--

Table 3 - Availability of Order Book Messages within Different Product States.

Please note that the BSE EOBI snapshot channels stop after migration of all products to “Post-End-Of-Day”.

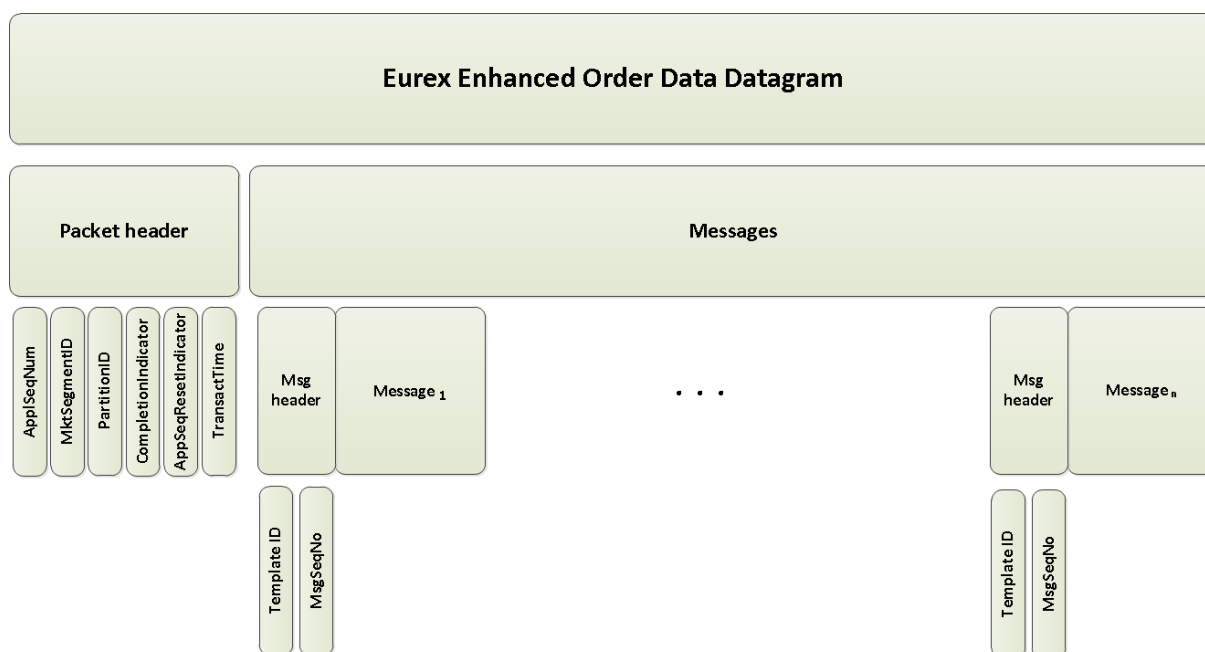
6 Message Formats

This chapter provides a global overview of the structure of datagram and message layouts and the data types used in these messages.

6.1 Datagram Structure

Each UDP datagram¹ starts with a Packet Header followed by one or more public market data messages and is terminated on the product level boundary, meaning that a datagram contains not more than order book updates for one product.

The BSE EOBI follows the following structure for the datagrams sent on the network:



Picture 3: Generic Datagram structure of BSE EOBI

The Packet Header in each datagram contains information about

- The product and the partition ID of corresponding product,
- A contiguous packet sequence number,
- An indicator whether the **atomic unit of work** fits into one datagram,
- An indicator whether a fail-over has occurred, and
- When the packet has been sent out.

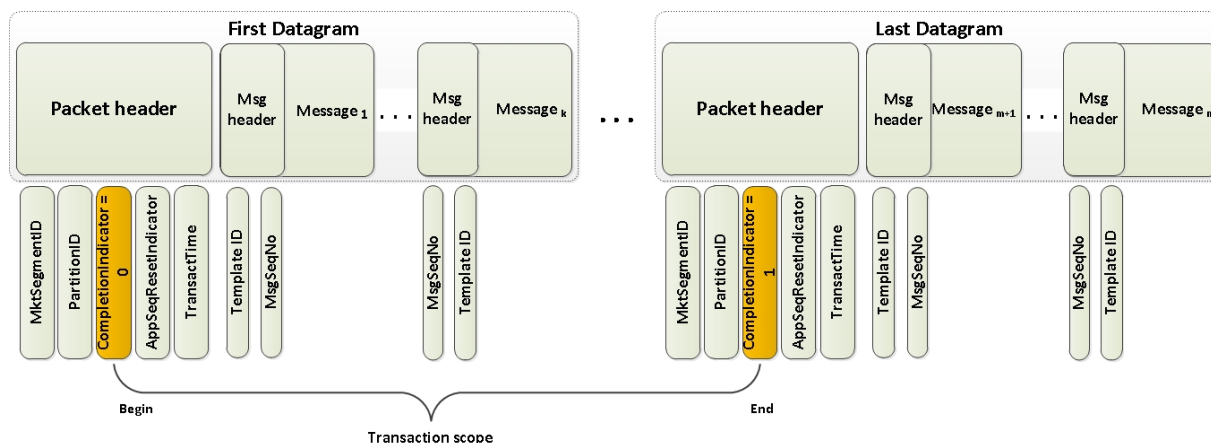
The product, *MarketSegmentID*, information can be used by participants for product filtering purposes.

The packet sequence numbers, *AppSeqNum*, are contiguous and are incremented per multicast address / port combination. They can be used by participants to detect gaps, duplicates and missing packets.

Furthermore, the Packet Header provides information whether the atomic unit of work that was processed by the corresponding matching engine fits into one datagram or is spread over several datagrams. By design, a datagram will contain one atomic unit of work that was

¹Shortly called a datagram to ease the readability.

processed by the corresponding matching engine. However, if the resulting public market data of one atomic unit of work doesn't fit into one datagram due to datagram size restriction, then the resulting market data information is spread over several datagrams. In this case, as it is shown in the picture below, the completion flag, i.e., *CompletionIndicator*, in the first packet header of the first datagram is set to *Incomplete* ($=0$) and in the packet header of the last datagram is set to *Complete* ($=1$). As a result, participants are able to gather all market data information belonging together.



Picture 4: Transaction scope spread over several datagrams

When the public market data fits into one datagram, the completion indicator in the packet header will be set to *Complete* ($=1$).

The time when the datagram has been sent out is provided by a timestamp, *TransactTime*.

The functional structure of each BSE EOBI datagram will always be the same; a message header will specify the fixed layout of the message content by a *templateID*, followed by a message sequence number of the corresponding product. Message sequence numbers, *MsgSeqNum*, contained in the market data incremental messages are incremented per product. Message sequence numbers for the market data snapshot messages are incremented per snapshot cycle.

The repeating groups in incremental and snapshot messages are not cut off.

6.2 Incremental Messages

Incremental messages are sent according to the BSE EOBI datagram structure as described above.

A message header will indicate the fixed layout of the message content, followed by the actual messages.

There is **no well-defined sending order** for the incremental messages. However, the *templateID* in the message header identifies each incremental message uniquely.

BSE EOBI incremental messages will be sent as long as the BSE EOBI service is available. The Heartbeat messages are repeated in the configured heartbeat interval in a single datagram by setting the message sequence number last sent to the *LastMsgSeqNumProcessed* field of the corresponding product. If the *LastMsgSeqNumProcessed* is not available, i.e., until the product state "Start-Of-Day", then it is set to "0".

As noted, if one atomic unit of work doesn't fit in one datagram, then the resulting market data information is spread over several datagrams. The completion flag will be used for this scenario.

Message	Template ID
Order Add	13100
Order Modify	13101
Order Modify Same Priority	13106
Order Delete	13102
Order Mass Delete	13103
Partial Order Execution	13105
Full Order Execution	13104
Execution Summary	13202
Auction Clearing Price	13501
Product State Change	13300
Instrument State Change	13301
Heartbeat	13001
Instrument Info	13203

Table 4 - BSE Enhanced Order Book incremental messages

For order book maintenance, the order messages Order Add, Order Modify, Order Delete and Order Mass Delete will be provided along with the product and instrument state messages. Execution for orders will be published via Partial Order Execution and Full Order Execution messages for partially and fully matched orders. Additionally, an execution summary, Execution Summary, message will be provided for the mass execution scenarios.

Functional Heartbeats will be published if there is no activity on a specific product.

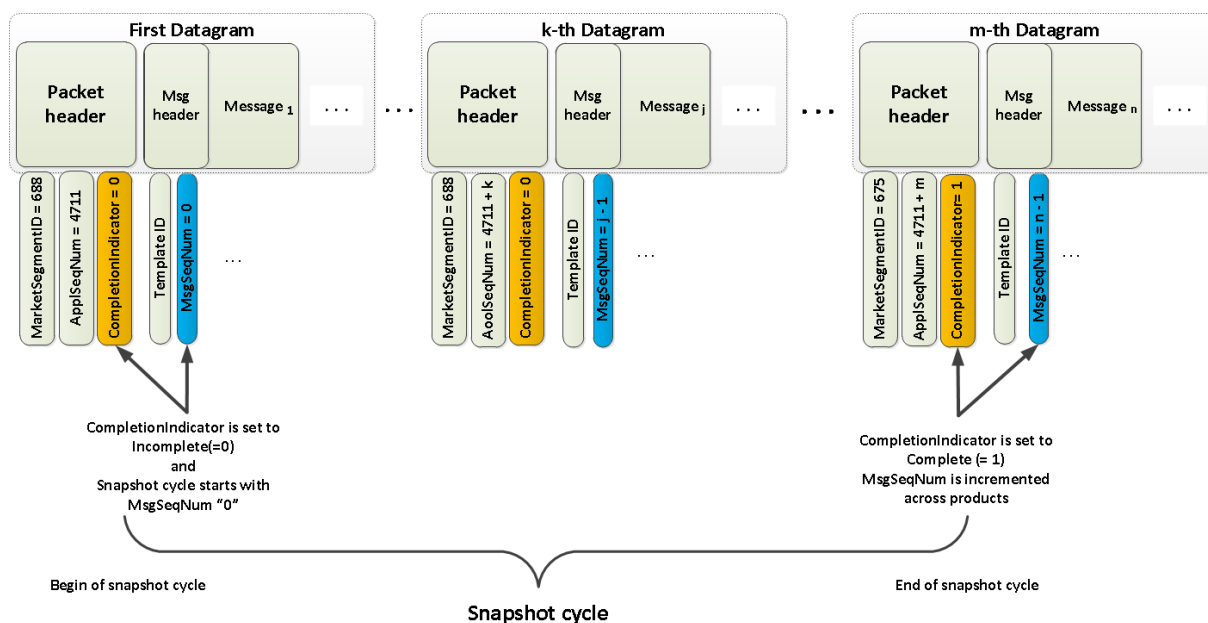
6.3 Snapshot Messages

By design, the snapshot messages are sent periodically and can be used by participants for recovery purposes, i.e., start-up processing or closing gaps in incremental messages. In contrast to BSE EOBI incremental messages, BSE EOBI snapshot messages will provide the trade statistics information at the time of sending. Furthermore, they contain the last message sequence number sent on the incremental feed, to provide a synchronization mechanism to participants for incremental and snapshots messages.

Like incremental messages, the snapshot messages will follow the BSE EOBI datagram structure as described in section 6.1 - "Datagram Structure".

BSE EOBI snapshot messages will be sent in product states between "Start-Of-Day" and "End-Of-Day".

The picture below outlines a typical **snapshot cycle**.



Picture 5: A snapshot cycle

It is characterized by,

- The packet sequence numbers, *ApplSeqNum*, are contiguous and are incremented across products,
- The message sequence number, *MsgSeqNum*, of the first message in the first datagram of a new snapshot cycle is set to zero(=0),
- The message sequence number, *MsgSeqNum*, within the same snapshot cycle is incremented for each message across all messages and all products,
- The *CompletionIndicator* in the last datagram of a product snapshot cycle is set to *Complete(=1)* to inform about the end of a product snapshot cycle.

That implies, a full snapshot cycle on BSE EOBI snapshot feed comprises of multiple product snapshot cycles. In order to assist an easy identification of a product snapshot boundary, the *CompletionIndicator* is set to *Complete(=1)* in the last datagram of a product. Each snapshot cycle starts by re-setting the message sequence number, *MsgSeqNum*, to zero(=0) for the first message in the first datagram.

Each message header containing the *templateID* of a message within a snapshot cycle will specify the message content.

Two summary messages are introduced to reduce the total size of snapshot messages in a snapshot cycle by avoiding redundant information:

- A Product Summary containing the last message sequence number of the last message sent on the incremental feed and trading state information, and
- An Instrument Summary for each instrument of the product including instrument state information and trade statistics such as last trade price and volume, daily low and high prices, opening prices etc. Additionally, the number of visible orders in the current product's snapshot cycle is provided to participants in advance.

The last message sequence number, *LastMsgSeqNumProcessed*, in the product summary message denotes the last message sent on the incremental feed, i.e., it provides a link between incremental and snapshot feed.

A snapshot cycle might contain order book information for multiple products. The following describes the snapshot cycle for one product.

A product has multiple instruments. The Product Summary will be given once, as it includes attributes that are identical for all instruments. However, it can include multiple Instrument Summary messages, each followed by the individual orders for that instrument.

As it shown in picture below, a **snapshot cycle of a product** will always start with a product summary followed by an instrument summary followed by all visible orders of the corresponding instrument and so on. Logically, the whole process is repeated for all instruments of a product.



Picture 6: A snapshot cycle of a product

Finally, as snapshot cycle of product is terminated on the product level boundary, i.e., *CompletionIndicator* is set to *Complete(=1)*, the next Product Summary message implicitly defines the start of a snapshot cycle for the next product, inherently defining the product level boundary. All messages within a product level boundary are self-consistent.

Order messages within a snapshot cycle will be sent in a zig-zag manner as described in 4.1 - "Building the Order Book". All subsequent products follow a similar pattern, forming a snapshot cycle.

BSE EOB snapshot messages will contain order book information about the intra-day created complex instruments as well, even if there is no trading activity in that complex instrument.

6.4 Message Overview

The following table provides a list of messages on the BSE EOBI in brief along with their classification. Furthermore, it is indicated during which instrument states the messages can be sent out.

6.5 Data Types

The following table provides an overview of the data types used in the fixed-length binary encoded messages sent out by the BSE EOBI. These data types will be indicated for each field in the Chapter 7.6 - "Message Layout", which covers the message layouts.

Data Type	Description	No Value
signed int.	little endian byte ordersupported are 1, 2, 4 and 8-byte,signed integers the most significant bit contains the sign.	1 byte signed int.: 0x802 byte signed int.: 0x80004 byte signed int.: 0x800000008 byte signed int.: 0x8000000000000000
unsigned int.	little endian byte ordersupported are 1, 2, 4 and 8-byte,unsigned integer.	1 byte unsigned int.: 0xFF2 byte unsigned int.: 0xFFFF4 byte unsigned int.: 0xFFFFFFFF8 byte unsigned int.: 0xFFFFFFFFFFFFFFFF
PriceType	Price in float format. For certain asset classes, prices may have negative values.	see 8 byte signed int.
Qty	Quantity in signed 4 byte integer format.	see 4 byte signed int.
UTCTimestamp	Date and time, in UTC, represented as nanoseconds past the UNIX epoch (00:00:00 UTC on January 1 st , 1970).	see 8 byte unsigned int.

Table 5 - Data types on the BSE EOBI

7 Message Layout

7.1 Overview of Supported Message Types

The following message formats are based on:

- Interface Version: 2.1
- Build Number: 502.001.201-502001200.224

General

EOBI Message	TemplateID (28500)	FIX Message	MsgType (35)
Packet Header	13002	MarketDataReport	U20
Heartbeat	13001	Heartbeat	0

Snapshot

EOBI Message	TemplateID (28500)	FIX Message	MsgType (35)
Product Summary	13600	MarketDataInstrument	U23
Snapshot Order	13602	MarketDataOrder	U21
Instrument Summary	13601	MarketDataInstrument	U23

Order Data

EOBI Message	TemplateID (28500)	FIX Message	MsgType (35)
Auction Clearing Price	13501	MarketDataInstrument	U23
Order Add	13100	MarketDataOrder	U21
Order Modify	13101	MarketDataOrder	U21
Order Modify Same Priority	13106	MarketDataOrder	U21
Order Delete	13102	MarketDataOrder	U21
Order Mass Delete	13103	MarketDataOrder	U21
Partial Order Execution	13105	MarketDataOrder	U21
Full Order Execution	13104	MarketDataOrder	U21

Trade Data

EOBI Message	TemplateID (28500)	FIX Message	MsgType (35)
Execution Summary	13202	MarketDataTrade	U22
Instrument Info	13203	MarketDataTrade	U22

Bombay Stock Exchange's Release 1.1

BSE India Pvt. Ltd

BSE's Enhanced Order Book Interface

V1.1

State Change

EOBI Message	TemplateID (28500)	FIX Message	MsgType (35)
Product State Change	13300	TradingSessionStatus	h
Instrument State Change	13301	SecurityStatus	f

7.2 General

Packet Header

The Packet Header is a technical header which is delivered in every UDP-datagram, and is used for identification of datagrams. The Packet Header will be published on a multicast-channel basis, with each packet containing information for one product only, recognizable by the field MarketSegmentID. Whenever there is an amount of information that doesn't fit in one datagram, the field CompletionIndicator will be set to 'Incomplete'. A CompletionIndicator field set to 'Incomplete' implies that another (new) datagram will follow, containing the missing data. This will be applied to the incremental messages only. Every partition stamps the outgoing datagrams with a sequence number: ApplSeqNum and a sending time: TransactTime. It also includes the ApplSeqResetIndicator field that can be set in case of market data fail-over and/or a market data restart (see Chapter 4.11).

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description						
<MessageHeader>												
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.						
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13002 (MarketDataReport, MsgType = U20)						
34	MsgSeqNum	U	4	4	unsigned int	not used						
<Message Body>												
1181	ApplSeqNum	Y	4	8	unsigned int	Message sequence number is contiguous and is incremented across products.						
1300	MarketSegmentID	Y	4	12	signed int	Product identifier.						
5948	PartitionID	Y	1	16	unsigned int	Grouping of BSE products. Belongs to the scope of Service Availability.						
6228	CompletionIndicator	Y	1	17	unsigned int	Indicated whether an unit of works fits into a single datagram for incremental messages. <table><tr><td>Value</td><td>Description</td></tr><tr><td>0</td><td>Incomplete</td></tr><tr><td>1</td><td>Complete</td></tr></table>	Value	Description	0	Incomplete	1	Complete
Value	Description											
0	Incomplete											
1	Complete											
28841	ApplSeqResetIndicator	Y	1	18	unsigned int	<table><tr><td>Value</td><td>Description</td></tr><tr><td>0</td><td>No Reset</td></tr><tr><td>1</td><td>Reset</td></tr></table>	Value	Description	0	No Reset	1	Reset
Value	Description											
0	No Reset											
1	Reset											
25020	Pad5	U	5	19	Fixed String	not used						
60	TransactTime	Y	8	24	UTCTimestamp	Time when market data feed handler writes packet on the wire.						

Implied Message Constants

These constant values are to be considered as part of the above message, although they are

Bombay Stock Exchange's Release 1.1

BSE India Pvt. Ltd

BSE's Enhanced Order Book Interface

V1.1

not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	U20	4	Fixed String	U20 = Market Data Report
28827	MDReportEvent	0	1	unsigned int	0 = Scope Definition.

Heartbeat

A functional Heartbeat message will be published regularly when there is no activity on the BSE Enhanced Order Book Interface snapshot or the BSE Enhanced Order Book Interface incremental channel. The functional Heartbeat message will be available on the BSE Enhanced Order Book Interface snapshot channel as of the trading state 'Pre-Trading' and will contain the last processed message sequence number, enabling participants to check for missed or lost packets. On the BSE Enhanced Order Book Interface incremental channel, the functional HeartBeat will be sent whenever the service is available.

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description
<MessageHeader>						
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13001 (Heartbeat, MsgType = 0)
34	MsgSeqNum	U	4	4	unsigned int	not used
<Message Body>						
369	LastMsgSeqNum-Processed	Y	4	8	unsigned int	Last Message Sequence number that was processed, regardless of message type.
25019	Pad4	U	4	12	Fixed String	not used

Implied Message Constants

These constant values are to be considered as part of the above message, although they are not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	0	4	Fixed String	0 = Hearbeat

7.3 Snapshot

Product Summary

A Product Summary message will be published once each snapshot cycle, and will contain attributes that are equal for all instruments that belong to that product.

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description	
<MessageHeader>							
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.	
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13600 (MarketDataInstrument, MsgType = U23)	
34	MsgSeqNum	Y	4	4	unsigned int	Message sequence number, incremented per product.	
<Message Body>							
369	LastMsgSeqNum-Processed	Y	4	8	unsigned int	Last Message Sequence number that was processed, regardless of message type.	
336	TradingSessionID	Y	1	12	unsigned int	Miscellaneous state information.	
						Value	Description
						1	Day
						3	Morning
						5	Evening
						7	Holiday
625	TradingSessionSubID	Y	1	13	unsigned int	Product state information.	
						Value	Description
						1	Pre Trading
						3	Trading
						4	Closing
						5	Post Trading
						6	Post Closing
						7	Quiescent
340	TradSesStatus	Y	1	14	unsigned int	Miscellaneous state information.	
						Value	Description
						1	Halted
						2	Open
						3	Closed

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description						
28828	FastMarketIndicator	Y	1	15	unsigned int	Indicates if product is in the state Fast Market. <table><tr><td>Value</td><td>Description</td></tr><tr><td>0</td><td>No</td></tr><tr><td>1</td><td>Yes</td></tr></table>	Value	Description	0	No	1	Yes
Value	Description											
0	No											
1	Yes											

Implied Message Constants

These constant values are to be considered as part of the above message, although they are not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	U23	4	Fixed String	U23 = Market Data Instrument
28842	MarketDataType	9	1	unsigned int	9 = Market Segment Snapshot

Snapshot Order

Each individual order or quote is represented in a Snapshot Order in a snapshot cycle on the BSE Enhanced Order Book Interface snapshot channel. The format of the snapshot order enables participants to build the order book according to price-time priority. For a more detailed view on a snapshot cycle see Chapters 4.1 and 6.3.

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description						
<MessageHeader>												
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.						
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13602 (MarketDataOrder, MsgType = U21)						
34	MsgSeqNum	Y	4	4	unsigned int	Message sequence number, incremented per product across all message types.						
<Message Body>												
<OrderDetails>												
21008	TrdRegTSTimePriority	Y	8	8	UTCTimestamp	Priority timestamp.						
1138	DisplayQty	Y	4	16	signed int	Quantity.						
54	Side	Y	1	20	unsigned int	Side of the order. <table><tr><td>Value</td><td>Description</td></tr><tr><td>1</td><td>Buy</td></tr><tr><td>2</td><td>Sell</td></tr></table>	Value	Description	1	Buy	2	Sell
Value	Description											
1	Buy											
2	Sell											
25018	Pad3	U	3	21	Fixed String	not used						
44	Price	Y	8	24	PriceType	Limit price. Required if OrdType (40) is Limit (2) or Stop Limit (4).						

Implied Message Constants

These constant values are to be considered as part of the above message, although they are not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	U21	4	Fixed String	U21 = Market Data Order
28842	MarketDataType	11	1	unsigned int	11 = Order Book Snapshot See also BSE EOBI Schema (XSD) file.
279	MDUpdateAction	5	1	unsigned int	5 = Overlay

Instrument Summary

An Instrument Summary message will be published for each instrument in one snapshot cycle on the BSE Enhanced Order Book Interface snapshot channel, and will contain instrument state information and trade statistics for one instrument. Note that one product can have multiple instruments. (See Figure 5 - BSE Enhanced Order Book Interface snapshot scenario.). The repeating group MDEntryGrp, instrument's trade statistics, are not cut off by design.

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description										
<MessageHeader>																
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.										
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13601 (MarketDataInstrument, MsgType = U23)										
34	MsgSeqNum	Y	4	4	unsigned int	Message sequence number, incremented per product.										
<Message Body>																
48	SecurityID	Y	8	8	signed int	Unique instrument identifier.										
779	LastUpdateTime	Y	8	16	UTCTimestamp	Last update time of the corresponding order book.										
21001	TrdRegTSExecution-Time	N	8	24	UTCTimestamp	Last matching execution timestamp.										
68	TotNoOrders	Y	2	32	Counter	Corresponding number of orders for this instrument.										
965	SecurityStatus	Y	1	34	unsigned int	Instrument status. <table><tr><th>Value</th><th>Description</th></tr><tr><td>1</td><td>Active</td></tr><tr><td>2</td><td>Inactive</td></tr><tr><td>4</td><td>Expired</td></tr><tr><td>9</td><td>Suspended</td></tr></table>	Value	Description	1	Active	2	Inactive	4	Expired	9	Suspended
Value	Description															
1	Active															
2	Inactive															
4	Expired															
9	Suspended															

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description	
326	SecurityTradingStatus	Y	1	35	unsigned int	Instrument state status.	
						Value	Description
						200	Closed
						201	Restricted
						202	Book
						203	Continuous
						204	Opening Auction
						205	Opening Auction Freeze
						206	Intraday Auction
						207	Intraday Auction Freeze
						208	Circuit Breaker Auction
						209	Circuit Breaker Auction Freeze
						210	Closing Auction
211	Closing Auction Freeze						
28828	FastMarketIndicator	Y	1	36	unsigned int	Value	Description
						0	No
						1	Yes
268	NoMDEntries	Y	1	37	Counter	Number of entries in Market Data message for MDEntryGrp.	
25017	Pad2	U	2	38	Fixed String	not used	
<MDInstrumentEntryGrp>						Variable size array, Record counter: NoMDEntries	
270	>MDEntryPx	N	8	40	PriceType	Price.	
271	>MDEntrySize	N	4	48	signed int	Quantity.	

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description	
269	>MDEntryType	Y	1	52	unsigned int	Type of market data entry.	
						Value	Description
						2	Trade
						4	Opening Price
						5	Closing Price
						7	High Price
						8	Low Price
						66	Trade Volume
						101	Previous Closing Price
						200	Opening Auction
						201	Intraday Auction
						202	Circuit Breaker Auction
						203	Closing Auction
						204	Upper Ckt Limit
						205	Lower Ckt Limit
206	Gross Trade Amt						
207	No Trades						
25018	>Pad3	U	3	53	Fixed String	not used	

Implied Message Constants

These constant values are to be considered as part of the above message, although they are not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	U23	4	Fixed String	U23 = Market Data Instrument
28842	MarketDataType	10	1	unsigned int	10 = Single Instrument Snapshot See also BSE EOBI Schema (XSD) file.
22	SecurityIDSource	M	1	Fixed String	M = Marketplace Marketplace assigned identifier.

7.4 Order Data

Auction Clearing Price

During auctions, no order book depth information is published. For a crossed order book in an auction, an Auction Clearing Price message will be published, indicating the potential auction price.

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description
<MessageHeader>						
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13501 (MarketDataInstrument, MsgType = U23)
34	MsgSeqNum	Y	4	4	unsigned int	Message sequence number, incremented per product across all message types.
<Message Body>						
60	TransactTime	Y	8	8	UTCTimestamp	Transaction timestamp.
48	SecurityID	Y	8	16	signed int	Unique instrument identifier.
31	LastPx	Y	8	24	PriceType	Indicating the potential Auction price for a crossed order book.
32	LastQty	Y	4	32	signed int	The potential matchable quantity for the auction.
25019	Pad4	U	4	36	Fixed String	not used

Implied Message Constants

These constant values are to be considered as part of the above message, although they are not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	U23	4	Fixed String	U23 = Market Data Instrument
28842	MarketDataType	6	1	unsigned int	6 = Auction Clearing Price See also BSE EOBI Schema (XSD) file.
22	SecurityIDSource	M	1	Fixed String	M = Marketplace Marketplace assigned identifier.

Order Add

An Order Add message will be published for each new order that was entered in the order book. The unique key for each order will be based on the instrument identifier, the priority timestamp and the order side, represented by the fields SecurityID, TrdRegTSTimePriority and Side respectively.

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description						
<MessageHeader>												
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.						
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13100 (MarketDataOrder, MsgType = U21)						
34	MsgSeqNum	Y	4	4	unsigned int	Message sequence number, incremented per product across all message types.						
<Message Body>												
21002	TrdRegTSTimeIn	N	8	8	UTCTimestamp	Matching engine In timestamp.						
48	SecurityID	Y	8	16	signed int	Unique instrument identifier.						
<OrderDetails>												
21008	TrdRegTSTimePriority	Y	8	24	UTCTimestamp	Priority timestamp.						
1138	DisplayQty	Y	4	32	signed int	Quantity.						
54	Side	Y	1	36	unsigned int	Side of the order. <table><tr><td>Value</td><td>Description</td></tr><tr><td>1</td><td>Buy</td></tr><tr><td>2</td><td>Sell</td></tr></table>	Value	Description	1	Buy	2	Sell
Value	Description											
1	Buy											
2	Sell											
25018	Pad3	U	3	37	Fixed String	not used						
44	Price	Y	8	40	PriceType	Price.						

Implied Message Constants

These constant values are to be considered as part of the above message, although they are not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	U21	4	Fixed String	U21 = Market Data Order
28842	MarketDataType	1	1	unsigned int	1 = Order Book Maintenance
279	MDUpdateAction	0	1	unsigned int	0 = New Type of Market Data update action.
22	SecurityIDSource	M	1	Fixed String	M = Marketplace Marketplace assigned identifier.

Order Modify

An Order Modify message will be published, if an existing order in the book is modified, whereby the new parameters of the order might cause a change in time priority. If an order is modified to another price, or if the quantity of this order is increased, the time priority of the order will change. The order that was modified is recognizable by the field TrdRegTSPrevTimePriority and a new priority key will be set by using the TrdRegTSTimePriority field.

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description						
<MessageHeader>												
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.						
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13101 (MarketDataOrder, MsgType = U21)						
34	MsgSeqNum	Y	4	4	unsigned int	Message sequence number, incremented per product across all message types.						
<Message Body>												
21002	TrdRegTSTimeIn	Y	8	8	UTCTimestamp	Matching engine In timestamp.						
21100	TrdRegTSPrevTimePriority	Y	8	16	UTCTimestamp	Previous order priority timestamp.						
28855	PrevPrice	Y	8	24	PriceType	Previous order price.						
28867	PrevDisplayQty	Y	4	32	signed int	Previous display quantity						
25019	Pad4	N	4	36	Fixed String	not used Valid characters: \x01-\x7E						
48	SecurityID	Y	8	40	signed int	Unique instrument identifier.						
<OrderDetails>												
21008	TrdRegTSTimePriority	Y	8	48	UTCTimestamp	Priority timestamp (new)						
1138	DisplayQty	Y	4	56	signed int	Quantity.						
54	Side	Y	1	60	unsigned int	Side of the order. <table><tr><td>Value</td><td>Description</td></tr><tr><td>1</td><td>Buy</td></tr><tr><td>2</td><td>Sell</td></tr></table>	Value	Description	1	Buy	2	Sell
Value	Description											
1	Buy											
2	Sell											
25018	Pad3	U	3	61	Fixed String	not used						
44	Price	Y	8	64	PriceType	Price.						

Implied Message Constants

These constant values are to be considered as part of the above message, although they are not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	U21	4	Fixed String	U21 = Market Data Order

Tag	Field Name	Field Value	Length	Data Type	Description
28842	MarketDataType	1	1	unsigned int	1 = Order Book Maintenance See also BSE EOBI Schema (XSD) file.
279	MDUpdateAction	1	1	unsigned int	1 = Change
22	SecurityIDSource	M	1	Fixed String	M = Marketplace

Order Modify Same Priority

An Order Modify Same Priority message will be published, if the time priority of an existing order is not changed. Please note that, if an order is modified to another price, or if the quantity of this order is increased, then the time priority of the order will change. Otherwise, the time priority remains as it is. The time priority of the order is available in the TrdRegTSTimePriority field.

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description						
<MessageHeader>												
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.						
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13106 (MarketDataOrder, MsgType = U21)						
34	MsgSeqNum	Y	4	4	unsigned int	Message sequence number, incremented per product across all message types.						
<Message Body>												
21002	TrdRegTSTimeIn	Y	8	8	UTCTimestamp	Matching engine In timestamp.						
60	TransactTime	Y	8	16	UTCTimestamp	Transaction timestamp.						
28867	PrevDisplayQty	Y	4	24	signed int	Previous display quantity						
25019	Pad4	N	4	28	Fixed String	Data structure padding (4 bytes). Valid characters: \x01-\x7E						
48	SecurityID	Y	8	32	signed int	Unique instrument identifier.						
<OrderDetails>												
21008	TrdRegTSTimePriority	Y	8	40	UTCTimestamp	Priority timestamp. Identical to the original time priority.						
1138	DisplayQty	Y	4	48	signed int	Quantity.						
54	Side	Y	1	52	unsigned int	Side of the order. <table><tr><td>Value</td><td>Description</td></tr><tr><td>1</td><td>Buy</td></tr><tr><td>2</td><td>Sell</td></tr></table>	Value	Description	1	Buy	2	Sell
Value	Description											
1	Buy											
2	Sell											
25018	Pad3	U	3	53	Fixed String	not used						
44	Price	Y	8	56	PriceType	Order Price.						

Implied Message Constants

These constant values are to be considered as part of the above message, although they are not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	U21	4	Fixed String	U21 = Market Data Order
28842	MarketDataType	1	1	unsigned int	1 = Order Book Maintenance
279	MDUpdateAction	1	1	unsigned int	1 = Change

Bombay Stock Exchange's Release 1.1

BSE India Pvt. Ltd

BSE's Enhanced Order Book Interface

V1.1

Tag	Field Name	Field Value	Length	Data Type	Description
22	SecurityIDSource	M	1	Fixed String	M = Marketplace Marketplace assigned identifier.

Order Delete

Whenever an existing order is deleted from the order book, an Order Delete message will be published. The Order Delete message will contain all necessary fields needed to delete the correct order; SecurityID, TrdRegTSTimePriority, Side. For convenience, the order delete message will also contain the former displayed quantity and the former price.

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description						
<MessageHeader>												
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.						
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13102 (MarketDataOrder, MsgType = U21)						
34	MsgSeqNum	Y	4	4	unsigned int	Message sequence number, incremented per product across all message types.						
<Message Body>												
21002	TrdRegTSTimeIn	N	8	8	UTCTimestamp	Matching engine In timestamp.						
60	TransactTime	Y	8	16	UTCTimestamp	Transaction timestamp.						
48	SecurityID	Y	8	24	signed int	Unique instrument identifier.						
<OrderDetails>												
21008	TrdRegTSTimePriority	Y	8	32	UTCTimestamp	Priority timestamp.						
1138	DisplayQty	Y	4	40	signed int	Quantity.						
54	Side	Y	1	44	unsigned int	Side of the order. <table><tr><th>Value</th><th>Description</th></tr><tr><td>1</td><td>Buy</td></tr><tr><td>2</td><td>Sell</td></tr></table>	Value	Description	1	Buy	2	Sell
Value	Description											
1	Buy											
2	Sell											
25018	Pad3	U	3	45	Fixed String	not used						
44	Price	Y	8	48	PriceType	Limit price. Required if OrdType (40) is Limit (2) or Stop Limit (4).						

Implied Message Constants

These constant values are to be considered as part of the above message, although they are not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	U21	4	Fixed String	U21 = Market Data Order
28842	MarketDataType	1	1	unsigned int	1 = Order Book Maintenance
279	MDUpdateAction	2	1	unsigned int	2 = Delete Type of Market Data update action.
22	SecurityIDSource	M	1	Fixed String	M = Marketplace Marketplace assigned identifier.

Order Mass Delete

An Order Mass Delete message will be published when the order book is expected to be emptied. The message contains the instrument identifier indicating which order book has to be fully deleted. Please note, that the Order Mass Delete Message is mutually exclusive to messages like TopOfBook, Auction Best Bid Offer and Auction Clearing Price which are published during non-continuous state. However it will be sent explicitly, if the potential auction price or BBO is not to be determined during auction call phase, i.e., neither Auction Best Bid Offer nor Auction Clearing Price messages are to be sent.

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description
<MessageHeader>						
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13103 (MarketDataOrder, MsgType = U21)
34	MsgSeqNum	Y	4	4	unsigned int	Message sequence number, incremented per product across all message types.
<Message Body>						
48	SecurityID	Y	8	8	signed int	Unique instrument identifier.
60	TransactTime	Y	8	16	UTCTimestamp	Transaction timestamp.

Implied Message Constants

These constant values are to be considered as part of the above message, although they are not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	U21	4	Fixed String	U21 = Market Data Order
28842	MarketDataType	1	1	unsigned int	1 = Order Book Maintenance See also BSE EOBI Schema (XSD) file.
279	MDUpdateAction	2	1	unsigned int	2 = Delete
22	SecurityIDSource	M	1	Fixed String	M = Marketplace

Partial Order Execution

Whenever a visible order is partially executed at its displayed price, a Partial Order Execution message will be published, containing the execution information; instrument identifier, priority timestamp, price and executed quantity of the executed passive order as well as the match identifier. The remaining quantity in the order book for this order must be calculated by subtracting the executed quantity in the Partial Order Execution message from the initial quantity in the order book.

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description						
<MessageHeader>												
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.						
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13105 (MarketDataOrder, MsgType = U21)						
34	MsgSeqNum	Y	4	4	unsigned int	Message sequence number, incremented per product across all message types.						
<Message Body>												
54	Side	Y	1	8	unsigned int	Side of the order. <table><tr><td>Value</td><td>Description</td></tr><tr><td>1</td><td>Buy</td></tr><tr><td>2</td><td>Sell</td></tr></table>	Value	Description	1	Buy	2	Sell
Value	Description											
1	Buy											
2	Sell											
25022	Pad7	U	7	9	Fixed String	not used						
44	Price	Y	8	16	PriceType	Reserved for future use. Currently, this price is equal to LastPx.						
21008	TrdRegTSTimePriority	Y	8	24	UTCTimestamp	Priority timestamp.						
48	SecurityID	Y	8	32	signed int	Unique instrument identifier.						
880	TrdMatchID	Y	4	40	unsigned int	Unique identifier for each price level (match step) of a match event; it is used for public trade reporting.						
32	LastQty	Y	4	44	signed int	Quantity executed in this fill.						
31	LastPx	Y	8	48	PriceType	Price of this fill.						

Implied Message Constants

These constant values are to be considered as part of the above message, although they are not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	U21	4	Fixed String	U21 = Market Data Order
28842	MarketDataType	2	1	unsigned int	2 = Order Book Execution
279	MDUpdateAction	1	1	unsigned int	1 = Change
22	SecurityIDSource	M	1	Fixed String	M = Marketplace

Full Order Execution

Whenever a visible order is fully executed at its displayed price, a Full Order Execution message will be published, containing the execution information; instrument identifier, priority timestamp, price and executed quantity of the executed passive order and the match identifier. As this order is executed in full, it has to be deleted from the order book.

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description						
<MessageHeader>												
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.						
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13104 (MarketDataOrder, MsgType = U21)						
34	MsgSeqNum	Y	4	4	unsigned int	Message sequence number, incremented per product across all message types.						
<Message Body>												
54	Side	Y	1	8	unsigned int	Side of the order. <table><tr><td>Value</td><td>Description</td></tr><tr><td>1</td><td>Buy</td></tr><tr><td>2</td><td>Sell</td></tr></table>	Value	Description	1	Buy	2	Sell
Value	Description											
1	Buy											
2	Sell											
25022	Pad7	U	7	9	Fixed String	not used						
44	Price	Y	8	16	PriceType	Reserved for future use. Currently, this price is equal to LastPx.						
21008	TrdRegTSTimePriority	Y	8	24	UTCTimestamp	Priority timestamp.						
48	SecurityID	Y	8	32	signed int	Unique instrument identifier.						
880	TrdMatchID	Y	4	40	unsigned int	Unique identifier for each price level (match step) of a match event; it is used for public trade reporting.						
32	LastQty	Y	4	44	signed int	Quantity executed in this fill.						
31	LastPx	Y	8	48	PriceType	Price of this fill.						

Implied Message Constants

These constant values are to be considered as part of the above message, although they are not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	U21	4	Fixed String	U21 = Market Data Order
28842	MarketDataType	2	1	unsigned int	2 = Order Book Execution
279	MDUpdateAction	1	1	unsigned int	1 = Change
22	SecurityIDSource	M	1	Fixed String	M = Marketplace

7.5 Trade Data

Execution Summary

Whenever an incoming order is executed, an Execution Summary message will be published, containing information on the execution of the incoming order. The Execution Summary message only contains information for the initial instrument (security), that was specified by the incoming order, i.e. any synthetic matches/changes can not be derived from the summary message. The Execution Summary message is meant for fast trading decisions only. In fact, to be absolutely sure the order book is correct, participants should always process the execution messages following the Execution Summary message. The fields in the Execution Summary message provide information on the instrument specified in the incoming order, the time the incoming order entered the matching engine, match time, the side of the incoming order, an indicator for a synthetic match, the quantity that was executed (of the specified instrument) in the fill, and the worst price of the fill, represented by the fields SecurityID, AggressorTimestamp, ExecID, AggressorSide, TradeCondition, LastQty, RestingHiddenQty and LastPx respectively. The RestingHiddenQty in the context of an execution would refer to the resting hidden quantity, included in the LastQty (of the specified instrument). It is set to zero if no such quantity is involved and is empty if the TradeCondition is flagged as ImpliedTrade.

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description						
<MessageHeader>												
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.						
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13202 (MarketDataTrade, MsgType = U22)						
34	MsgSeqNum	Y	4	4	unsigned int	Message sequence number, incremented per product across all message types.						
<Message Body>												
48	SecurityID	Y	8	8	signed int	Unique instrument identifier.						
28820	AggressorTimestamp	N	8	16	UTCTimestamp							
17	ExecID	Y	8	24	UTCTimestamp	Matching timestamp.						
32	LastQty	Y	4	32	signed int	Total quantity of this match.						
28731	AggressorSide	Y	1	36	unsigned int	<table><tr><th>Value</th><th>Description</th></tr><tr><td>1</td><td>Triggered by the buy side</td></tr><tr><td>2</td><td>Triggered by the sell side</td></tr></table>	Value	Description	1	Triggered by the buy side	2	Triggered by the sell side
Value	Description											
1	Triggered by the buy side											
2	Triggered by the sell side											
277	TradeCondition	N	1	37	unsigned int	<div>Indicates whether a synthetic match is occurred.</div> <table><tr><th>Value</th><th>Description</th></tr><tr><td>1</td><td>Implied Trade</td></tr></table>	Value	Description	1	Implied Trade		
Value	Description											
1	Implied Trade											
25017	Pad2	U	2	38	Fixed String	not used						
31	LastPx	Y	8	40	PriceType	Worst price of this match.						

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description
28868	RestingHiddenQty	N	4	48	signed int	Quantity of matched passive orders that is not displayed to the market. Set to zero if no such quantity is involved. Is empty if the TradeCondition is flagged as ImpliedTrade.
25019	Pad4	U	4	52	Fixed String	not used

Implied Message Constants

These constant values are to be considered as part of the above message, although they are not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	U22	4	Fixed String	U22 = Market Data Trade
28842	MarketDataType	12	1	unsigned int	12 = Match Event See also BSE EOBI Schema (XSD) file.
279	MDUpdateAction	0	1	unsigned int	0 = New Type of Market Data update action.
22	SecurityIDSource	M	1	Fixed String	M = Marketplace Marketplace assigned identifier.

Instrument Info

Instrument Info – An Instrument Info message will be published for an instrument on the BSE Enhanced Order Book Interface incremental channel whenever there is a change in the daily price range of the instrument.

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description
<MessageHeader>						
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13203 (MarketDataTrade, MsgType = U22)
34	MsgSeqNum	Y	4	4	unsigned int	Message sequence number, incremented per product across all message types.
<Message Body>						
48	SecurityID	Y	8	8	signed int	Unique instrument identifier.
5	ClosePrice	N	8	16	PriceType	Close Price
140	PrevClosePrice	N	8	24	PriceType	Previous Close Price
332	UpperCktLimit	N	8	32	PriceType	Upper Circuit Limit Price
333	LowerCktLimit	N	8	40	PriceType	Lower Circuit Limit Price

Implied Message Constants

These constant values are to be considered as part of the above message, although they are not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	U22	4	Fixed String	Defines message type ALWAYS FIRST FIELD IN MESSAGE. (Always unencrypted) Note: A 'U' as the first character in the MsgType field (i.e. U, U2, etc) indicates that the message format is privately defined between the sender and receiver.
28842	MarketDataType	14	1	unsigned int	Type of public market data, e.g., Order Book Maintenance (=1), Order Book Execution (=2), Market Segment Snapshot (=9) etc. Valid values are available in the BSE EOBI Schema (XSD) file.

7.6 State Change

Product State Change

The Product State Change message provides updates on the trading state for (all instruments in) a particular product.

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description	
<MessageHeader>							
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.	
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13300 (TradingSessionStatus, MsgType = h)	
34	MsgSeqNum	Y	4	4	unsigned int	Message sequence number, incremented per product across all message types.	
<Message Body>							
336	TradingSessionID	Y	1	8	unsigned int	Miscellaneous state information.	
						Value	Description
						1	Day
						3	Morning
						5	Evening
						7	Holiday
625	TradingSessionSubID	Y	1	9	unsigned int	Product state information.	
						Value	Description
						1	Pre Trading
						3	Trading
						4	Closing
						5	Post Trading
						6	Post Closing
7	Quiescent						
340	TradSesStatus	Y	1	10	unsigned int	Miscellaneous state information.	
						Value	Description
						1	Halted
						2	Open
	3	Closed					
28828	FastMarketIndicator	Y	1	11	unsigned int	Indicates if product is in the state Fast Market.	
						Value	Description
						0	No
						1	Yes
25019	Pad4	U	4	12	Fixed String	not used	

Bombay Stock Exchange's Release 1.1

BSE India Pvt. Ltd

BSE's Enhanced Order Book Interface

V1.1

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description
60	TransactTime	Y	8	16	UTCTimestamp	Transaction timestamp.

Implied Message Constants

These constant values are to be considered as part of the above message, although they are not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	h	4	Fixed String	h = Trading Session Status
1368	TradSesEvent	3	1	unsigned int	3 = Status Change

Instrument State Change

The Instrument State Change message provides state information for a single instrument. Furthermore, it informs participants about intra-day expiration of instruments.

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description																										
<MessageHeader>																																
9	BodyLen	Y	2	0	unsigned int	Number of bytes for the message, including this field.																										
28500	TemplateID	Y	2	2	unsigned int	Unique identifier for a BSE EOBI message layout. Value: 13301 (Security-Status, MsgType = f)																										
34	MsgSeqNum	Y	4	4	unsigned int	Message sequence number, incremented per product across all message types.																										
<Message Body>																																
48	SecurityID	Y	8	8	signed int	Unique instrument identifier.																										
965	SecurityStatus	Y	1	16	unsigned int	Instrument status. <table><tr><td>Value</td><td>Description</td></tr><tr><td>1</td><td>Active</td></tr><tr><td>2</td><td>Inactive</td></tr><tr><td>4</td><td>Expired</td></tr><tr><td>9</td><td>Suspended</td></tr></table>	Value	Description	1	Active	2	Inactive	4	Expired	9	Suspended																
Value	Description																															
1	Active																															
2	Inactive																															
4	Expired																															
9	Suspended																															
326	SecurityTradingStatus	Y	1	17	unsigned int	Instrument state status. <table><tr><td>Value</td><td>Description</td></tr><tr><td>200</td><td>Closed</td></tr><tr><td>201</td><td>Restricted</td></tr><tr><td>202</td><td>Book</td></tr><tr><td>203</td><td>Continuous</td></tr><tr><td>204</td><td>Opening Auction</td></tr><tr><td>205</td><td>Opening Auction Freeze</td></tr><tr><td>206</td><td>Intraday Auction</td></tr><tr><td>207</td><td>Intraday Auction Freeze</td></tr><tr><td>208</td><td>Circuit Breaker Auction</td></tr><tr><td>209</td><td>Circuit Breaker Auction Freeze</td></tr><tr><td>210</td><td>Closing Auction</td></tr><tr><td>211</td><td>Closing Auction Freeze</td></tr></table>	Value	Description	200	Closed	201	Restricted	202	Book	203	Continuous	204	Opening Auction	205	Opening Auction Freeze	206	Intraday Auction	207	Intraday Auction Freeze	208	Circuit Breaker Auction	209	Circuit Breaker Auction Freeze	210	Closing Auction	211	Closing Auction Freeze
Value	Description																															
200	Closed																															
201	Restricted																															
202	Book																															
203	Continuous																															
204	Opening Auction																															
205	Opening Auction Freeze																															
206	Intraday Auction																															
207	Intraday Auction Freeze																															
208	Circuit Breaker Auction																															
209	Circuit Breaker Auction Freeze																															
210	Closing Auction																															
211	Closing Auction Freeze																															
28828	FastMarketIndicator	Y	1	18	unsigned int	<table><tr><td>Value</td><td>Description</td></tr><tr><td>0</td><td>No</td></tr><tr><td>1</td><td>Yes</td></tr></table>	Value	Description	0	No	1	Yes																				
Value	Description																															
0	No																															
1	Yes																															
25020	Pad5	U	5	19	Fixed String	not used																										

Bombay Stock Exchange's Release 1.1

BSE India Pvt. Ltd

BSE's Enhanced Order Book Interface

V1.1

Tag	Field Name	Req'd	Len	Ofs	Data Type	Description
60	TransactTime	Y	8	24	UTCTimestamp	Transaction timestamp.

Implied Message Constants

These constant values are to be considered as part of the above message, although they are not transmitted.

Tag	Field Name	Field Value	Length	Data Type	Description
35	MsgType	f	4	Fixed String	f = Security Status
22	SecurityIDSource	M	1	Fixed String	M = Marketplace Marketplace assigned identifier.

8 Appendix

8.1 Product Scope

BSE EOBI interface is designed for all BSE products.

8.2 Synthetic prices

The synthetic matching for futures products, i.e., for futures spread instruments, can be enabled for synthetic matching. Any incoming order can match synthetically against the order books enabled for synthetic matching. Information about whether a futures spread instrument is enabled (*ImpliedMarketIndicator*) for synthetic matching, is published by public reference data in both in the Instrument Snapshot message and in the Complex Instrument Update message.

BSE Functional Reference documentation describes the synthetic matching for futures spread instruments that are enabled for synthetic matching in detail. Furthermore in this document, the terms such as Match Path, Synthetic Book Path, Synthetic Pricing and available Quantity referring to Synthetic Book Path, are explained thoroughly. Additionally, the general rules for the calculation of a synthetic price are explained in-depth.

For more details please see paragraph about Synthetic Matching in Continuous Trading.

8.3 Connecting BSE EOBI and BSE ETI data

BSE EOBI and BSE ETI provide information to synchronize private responses and public market data.

Order transactions

The Order time priority and matching engine in timestamp information is provided by both interfaces, i.e., *ExecID* field from BSE ETI in Order Status and Execution Reports and *TrdRegTSTimePriority* field from BSE EOBI in incremental and snapshot messages along with the *securityID* information as shown in the table below:

Field Description	Public Market Data via BSE EOBI	Private Market Data via BSE ETI
Security Identifier	<i>securityID</i>	<i>securityID</i>
Priority Timestamp of an Order	<i>TrdRegTSTimePriority</i>	<i>TrdRegTSTimePriority</i> / <i>ExecID</i> for Standard orders <i>ExecID</i> for Lean orders
Matching Engine In Timestamp	<i>TrdRegTSTimeIn</i>	<i>TrdRegTSTimeIn</i>
Match Step Identifier	<i>TrdMatchID</i>	<i>FillMatchID</i> <i>QuoteEventMatchID</i> <i>TrdMatchID</i>

Table 8 – Provided private and public data via interfaces

An order that is modified will lose its time priority, i.e., it will get a new priority time stamp, if its price or its quantity or order type is changed. So,

Order executions

When an order executes against the order book at multiple price levels, this is reflected by a matching event with multiple match steps. Each match step includes the trades at one price level and is represented by a unique *TrdMatchID* (880) and published in the public market data.

The field *TrdMatchID* (880) is a unique id on product level for each business day. A synthetic match can result in more than one trade volume record with the same *TrdMatchID* (880). Every match step occurring in the exchange has an identifier in BSE ETI that is provided in the field *FillMatchID* (28708) in the Execution Report (8), *QuoteEventMatchID* (8714) in the Quote Execution Report (U8) and *TrdMatchID* (880) in the Trade Capture Report (AE). The match time of all involved orders is reported in the Execution Summary message by using the field *ExecID* (17). These identifier allows participants to link trade capture reports and the corresponding execution report of the BSE ETI with the market data incremental feed of the BSE EOBI.

The aggressor timestamp is always identical to the matching engine in timestamp.

8.4 Multicast addresses

The reference information provided by BSE RDI contains the respective multicast channel information (i.e., multicast addresses and port numbers) for all available products.

Reference data snapshot channels

Environment	Service A	Service B
Production		
Simulation		

Table 9 – Multicast address and ports for reference data snapshot feeds

Reference data incremental channels

Environment	Service A	Service B
Production		
Simulation		

Table 10 – Multicast address and ports for reference data incremental feeds

Please note that the reference data is provided in file form as compressed Reference Data Files (**RDF**) in FIXML-layout, updated approximately every 5 minutes via the Common Report Engine(CRE).

8.5 Reference data for BSE EOBI

The reference data information such as order book type, multicast addresses and port numbers of corresponding products etc., which is needed to receive public market data via BSE EOBI, is available via the existing BSE RDI and/or the BSE RDF.

The Product Snapshot message will contain the following information for the products configured for BSE EOBI:

- Book Type, *MDBookType(1021)*, field will carry the valid value (*Order Depth = 3*),
- Feed Type, *MDFeedType(1022)*, field will carry the valid values (*HI* = high bandwidth incrementals) for BSE EOBI incremental messages and (*HS* = high bandwidth snapshots) for BSE EOBIsnapshot messages in combination with the multicast addresses and port numbers,
- IP Multicast address, *MDPrimaryFeedLineID(28590)*, field will carry the IP Multicast address of primary BSE EOBI feed along with the primary port number, *MDPrimaryFeedLineSubID(28591)*.
- IP Multicast address, *MDSecondaryFeedLineID(28590)*, field will carry the IP Multicast address of secondary BSE EOBI feed along with the secondary port number (*MDSecondaryFeedLineSubID(28591)*).

The same information is also available via BSE Reference Data Files (BSE RDF).

9 Change log

The updated BSE EOBI interface version, '2.1' (as part of BSE Exchange's T7 release 2.1).

Chapter	Description
All	Implied message constants made available along with the EOBI message layouts Description of Heartbeat message is improved.
4.1, 5, 4.13, 7.4, 7.5, 8.2	Description of the sending order sequence in zig-zag fashion is improved by an example. Transaction timestamp is added in the Auction messages. <i>TradeCondition</i> is added in the Execution Summary message. Description of Recovery handling is improved. Snapshot Order message is not sent during Pre-Trading phases. IP Multicast address range table is added. Description of implied message constants have been improved in the message layouts. Description of Order Modify is improved.
7.4	The previous fields in Order Modify message are mandatory.
4.11, 7.4	FIX Tag ID of <i>TransactTime</i> is changed from 273 to 60. Description of Partial Order Execution message is improved.
4, 7.4, 8.2	Description of Order Mass Delete message is improved. It is sent explicitly, if the potential auction price is not determined during auction call phase. A new sub-chapter Synthetic prices with a reference to Functional Reference document is added. Description of Partial Order Execution message is improved.
4, 4.4, 4.5, 5, 6.2, 6.3, 7.1, 7.4	<i>CompletionIndicator</i> set to Complete(=1) on the last datagram of every Product Snapshot cycle on BSE EOBI snapshot feed. Description of Order Mass Delete message is improved. It is sent explicitly, if the auction bbo or auction clearing price is not to be determined during auction call phase. Added new message description and layout for OrderModifySamePrio. Added <i>TransactTime</i> to OrderDelete, OrderMassDelete and OrderModifySamePrio messages. Updated packet header <i>TemplateID</i> to 13002.
3.2, 4.6, 7.5	The MTU size is set to 1372 bytes. Added field <i>RestingHiddenQty</i> to Execution Summary message.
4.2, 7.5	The field <i>TrdRegTSTimeIn</i> is not mandatory in OrderAdd and OrderDelete messages.
7.3, 7.6	An additional enum Post closing added for the field <i>TradingSessionSubID</i> . New Enum added for field <i>MDEntryType</i> in Instrument Snapshot message.
7.5	A new message Instrument Info added to communicate any intra-day change in Daily Price range (circuit limits) of the instrument.
7.4	A new field <i>LastQty</i> added to provide indicative auction quantity .